AMENDMENTS TO THE CLAIMS

1.   (Currently Amended) A computer implemented method for dynamically

composing and maintaining applications over a computer architecture comprising:

receiving an indication to dynamically integrate a component into an executing

application, ~~wherein the component includes a new component to replace~~

~~an existing component~~;

loading the component from a source;

linking the component to the application in runtime by obtaining an integration

interface associated with the component, the integration interface

providing methods for managing the component, and by initializing the

component by invoking an initialize method of the integration interface;

and

~~invoking an initialize method to initialize the component; and~~

establishing an inter-component communication between the component and

existing components by

publishing first information associated with the loaded component for the

existing components,

subscribing second information associated with the existing components

by the loaded component, and

removing previously published interfaces, the previously published

interfaces are removed by the loaded component.

~~invoking a replace method to transition an existing state of the existing~~

~~component into the new component.~~

2.   (Cancelled)

3.	(Currently Amended) The method of claim 1, ~~further comprising supporting the component's ability to allow other components to communicate with it by:~~ wherein the publishing of the first information is performed by invoking a publish method of the integration interface, the first information including publishing one or more of data, events, and ~~specifying~~ one or more interfaces ~~to publish~~ to ~~other~~ the existing components; ~~and~~

~~storing the one or more interfaces in an interface clearinghouse~~.

4.	(Currently Amended) The method of claim 3, ~~further comprising supporting the component's ability to communicate with other components by:~~ ~~consulting the interface clearinghouse to deter~~mine one or more interfaces to ~~retrieve from other component;~~ wherein the subscribing of the second information is performed by invoking a retrieve method of the integration interface, the subscribing of the second information including retrieving ~~and specifying an interface of the~~ one or more interfaces ~~to retrieve~~ from ~~other component~~ the existing components, the subscribing of the second information further including the loaded component communicating with the existing components via; ~~and~~ using the retrieved ~~interface~~ one or more interfaces ~~to communicate with other components~~.

5.	(Currently Amended) The method of claim 1, further comprising:
examining the existing components to determine a defective component that is
~~invoking a stop method of the integration interface when the component is~~ ~~ready~~ to be ~~shut down~~ replaced; and
replacing the defective component with the loaded component.

Attorney Docket No. 42390P12319		3
Application No. 10/040,654

6.   (Currently Amended) The method of claim 1, wherein the ~~application resides in a~~ ~~network, and the loading of the component~~ source comprises ~~retrieving the~~ ~~component from a member in the~~ one or more of a file system and a component repository~~network~~.

7.   (Cancelled)

8.   (Currently Amended) The method of claim ~~7~~ 6, wherein the ~~peer comprises~~ ~~another component loader in the~~ file system and the component repository reside in a network.

Claims 9-13 (Cancelled)

14.   (Currently Amended) A system comprising:

a storage device;

a client computer system coupled with the storage device; and

a server computer system coupled with the client computer system, the server

computer system including a component framework to dynamically

compose and maintain applications over a computer architecture, the

component framework is further to

receiving an indication to dynamically integrate a component into an

existing application, ~~an integration interface providing methods for~~

~~managing a component, the methods including an initialize method~~

~~to transition a given component into a state to operate, a replace~~

~~method to transition an existing state of an existing component into~~

~~a new component, and a stop method to transition the given~~

~~component into a destroy state,~~

~~at least one component that implements the integration interface,~~

a components repository for storing the at least one component,

a communications bus, wherein the communication bus is established after

at least one call to a publish method and a retrieve method of the

integration interface, and

a component framework corresponding to an application to manage the at

least one component using the integration interface, the component

framework having

a component loader to load the component from a source, the source

including one or more of a file system and a component requested

components from the components repository, into an application,

an interface clearinghouse to store and manage interfaces

corresponding to the loaded components, and

a messaging mechanism for external entities to communicate with

the loaded components.

link the component to the application in runtime by obtaining an

integration interface associated with the component, the integration

interface providing methods for managing the component, and by

initializing the component by invoking an initialize method of the

integration interface, and

establish an inter-component communication between the component and

existing components by

publishing first information associated with the loaded component

for the existing components,

> subscribing second information associated with the existing
>
> components by the loaded component, and
>
> removing previously published interfaces, the previously published
>
> interfaces are removed by the loaded component.

15. (Currently Amended) The system of claim 14, wherein the inter-component communication is established via a communication bus ~~is to facilitate inter-component communication~~.

16. (Cancelled)

17. (Currently Amended) The system of claim 14, wherein the component framework is further to:~~application resides in a network, and the loading of the component comprises retrieving the component from a member in the network~~

> examine the existing components to determine a defective component that is to be
>> replaced; and
>
> replace the defective component with the loaded component.

18. (Currently Amended) A machine-readable medium having instructions which, when executed, cause a machine to:

    receive an indication to dynamically integrate a component into an executing

>> application, ~~wherein the component is a new component~~;

    load the component from a source;

    link the component to the application in runtime by obtaining the component's

>> integration interface, the integration interface providing methods for
>>
>> managing the component, and by initializing the component by invoking
>>
>> an initialize method ~~to initialize the component~~ of the integration
>>
>> interface; and

establish an inter-component communication between the component and existing

components by

publishing first information associated with the loaded component for the

existing components,

subscribing second information associated with the existing components

by the loaded component, and

removing previously published interfaces, the previously published interfaces are

removed by the loaded component.

~~replace an existing component by invoking a replace method to transition~~

~~an existing state of the existing component into the new~~

~~component.~~

19.   (Cancelled)

20.   (Currently Amended) The machine-readable medium of claim 18, wherein the

~~instructions when further executed, cause the machine to support the component's~~

~~ability to allow other components to communicate with it by:~~ publishing of the

first information is performed by invoking a publish method of the integration

interface, the first information including publishing one or more of data, events,

and ~~specifying~~ one or more interfaces to ~~publish to other~~ the existing components;

and storing the one or more interfaces in an interface clearinghouse.

21.   (Currently Amended) The machine-readable medium of claim 20, wherein the

~~instructions when further executed, cause the machine to support the component's~~

~~ability to communicate with other components by:~~ subscribing of the second

information is performed by ~~consulting the interface clearinghouse to determine~~

~~one or more interfaces to retrieve from other component;~~ invoking a retrieve

method of the integration interface, the subscribing of the second information including retrieving ~~and specifying an interface of the one or more interfaces to retrieve from other component; and using the retrieved interface to communicate with other components~~ from the existing components, the subscribing of the second information further including the loaded component communicating with the existing components via the retrieved one or more interfaces.

22.   (Currently Amended) The machine-readable medium of claim 18, wherein the instructions when executed, further cause the machine to:

~~application resides in a network, and the loading of the component comprises retrieving the component from a member in the network.~~

examine the existing components to determine a defective component that is replaced; and

replace the defective component with the loaded component.

Claims 23-30 (Cancelled)